REMARKS

Reconsideration of the application in view of the above amendments and the following remarks is respectfully requested. Claims 1, 3, 5, 8-23, 25, 27, and 30-33 have been amended. Claims 1-33 are currently pending in the application.

CLAIM REJECTIONS – 35 U.S.C. §101

In the Office Action, the Examiner rejected claims 12-22 under 35 U.S.C. §101 for being directed to non-statutory subject matter. Without any admission as to the veracity of the Examiner's rejection, but rather in the interest of advancing prosecution, Applicants have amended claims 12-22 to replace each instance of "machine-readable medium" with "machine-readable storage medium". Applicants believe that this amendment addresses the Examiner's concerns. Hence, Applicants request that this rejection be withdrawn.

The Examiner also rejected claims 23-33 under 35 U.S.C. §101. As justification for the rejection, the Examiner stated that, even though the claims recited an apparatus, the apparatus would be reasonably interpreted by one of ordinary skill in the art as software per se, thus, failing to be tangibly embodied or include any recited hardware. Applicants respectfully disagree with this rationale. As they currently stand, claims 23-33 recite <u>mechanisms</u> for performing certain functions. As is well known by those of ordinary skill in the computing arts, software per se is not a <u>mechanism</u>. By itself, software does not perform any function. Rather, software has to be executed by one or more processors in order for any functions to be realized. This fact is extremely well known in the computing arts. Consequently, one of ordinary skill in the computing arts <u>would not</u> reasonably interpret the mechanisms and the apparatus of claims 23-33 to be software per se. Accordingly, Applicants respectfully request that this rejection be withdrawn.

CLAIM REJECTIONS – 35 U.S.C. §102 and §103

In the Office Action, the Examiner rejected claims 1-4, 7-8, 12-15, 18-19, 23-26 and 29-30 under 35 U.S.C. §102(e) as being anticipated by Armstrong et al. (U.S. Publication No. US 2002/0156824 A1). The Examiner also rejected claims 5-6, 9-11, 16-17, 20-22, 27-28, and 31-33 under 35 U.S.C. §103(a) as being unpatentable over Armstrong et al. In response to these rejections, independent claims 1, 12, and 23 have been amended to clarify the subject matter that is being claimed.

Claim 1

Claim 1 has been amended to recite:

> A method performed by an operating system, comprising:
> establishing a plurality of non-global operating system partitions within a global operating system environment provided by the operating system, wherein each non-global operating system partition serves to isolate processes running within that non-global operating system partition from other non-global operating system partitions within the global operating system environment, wherein enforcement of boundaries between the non-global operating system partitions is carried out by the operating system, and wherein the plurality of non-global operating system partitions comprises a particular non-global operating system partition;
> associating the particular non-global operating system partition with a first resource pool comprising one or more resources; and
> ensuring that processes running within the particular non-global operating system partition are allowed to utilize only the resources in the first resource pool. (Emphasis added)

Claim 1 has been amended to point out that the method is performed by an operating system, so that establishing, associating, and ensuring are all performed by the operating system. Claim 1 also points out that a plurality of non-global operating system partitions are established within a global operating system environment provided by the operating system. Claim 1 further makes it clear that the boundaries between the non-global operating system partitions are enforced by the operating system. These amendments are amply supported by the Specification (see e.g. Fig. 1, paragraphs 0003, 0010-0011, 0015, 0018, etc.).

Such a method is neither disclosed nor suggested by Armstrong. Instead, Armstrong discloses a mechanism for enabling a computer system to be partitioned into a plurality of logical partitions. The system of Armstrong comprises a low-level hypervisor base 202 (Fig. 2 of Armstrong), which executes at a level beneath the OS kernel level. The hypervisor base 202 enables the system to be logically partitioned. Specifically, it is the hypervisor base 202, along with a hypervisor management tool 203, that enables an administrator to configure the system into a plurality of logical partitions 204A-204D (paragraph 0025), and it is the hypervisor base 202 that enforces the logical partitions (paragraph 0034, lines 1-4). Each partition may have a set of resources (e.g. processors) associated therewith, and each partition may execute its own OS kernel (paragraph 0026, last sentence; Fig. 2). Set up in this way, each partition behaves like a separate and distinct computer system. Thus, with the mechanism of Armstrong, a single computer system can be partitioned to behave like a plurality of separate and distinct systems.

Several distinguishing points should be noted with regard to Armstrong. First of all, note that in Armstrong, it is the hypervisor (made up of portions 202 and 203), not the OS kernels, that establishes the logical partitions 204A-204D. As stated in paragraph 0037, lines 1-9, an administrator uses hypervisor portion 203 to create and/or alter logical partition definitions. Once the logical partitions are defined, the "hypervisor causes state values to be written to various hardware registers and other structures, which define the boundaries and behavior of the logical partitions" (paragraph 0037, last sentence). From these excerpts, it is clear that it is the hypervisor, which executes beneath the OS kernel level, that establishes the logical partitions 204A-204D. In contrast to the method of claim 1, it is not an operating system that establishes the partitions. There is no teaching whatsoever in Armstrong of having one of the OS kernels establish the partitions; thus, this aspect of claim 1 is clearly not taught or suggested by Armstrong.

Another point to note is that, because the logical partitions 204A-204D in Armstrong are not established by an operating system, they are not operating system partitions. Put another way, they are not partitions established by an operating system within an operating system environment provided by the operating system. If they were, the partitions would look like the partitions shown in Fig. 1 of the present application, wherein a plurality of non-global partitions 104 are shown within a global operating environment 130 provided by an operating system. There is no such showing in Armstrong. Instead, in Fig. 2 of Armstrong, each of the partitions 204A-204D is shown as a separate partition, each of the partitions is shown as executing a separate OS kernel, and none of the OS kernels show multiple partitions within it. Hence, unlike claim 1, the logical partitions of Armstrong are not operating system partitions, and they are not established by an operating system within an operating system environment provided by the operating system. This aspect of claim 1 is clearly not shown or suggested by Armstrong.

Yet another point to note is that in Armstrong, it is the hypervisor 202, not the OS kernels, that enforces the boundaries between the logical partitions. This is made clear in the first sentence of paragraph 0034, which states: "Immediately above the hardware is a common low-level hypervisor base 202, also called partitioning licensed internal code (PLIC), which enforces logical partitioning". Thus, in contrast to claim 1 in which it is the operating system that enforces the boundaries between operating system partitions, the system of Armstrong does not employ any OS kernel (but rather, employs the hypervisor) to perform the boundary enforcement function. This aspect of claim 1 is neither disclosed nor suggested by Armstrong.

Yet another point to note is that in Armstrong, it is the hypervisor 202, not the OS kernels, that ensures that tasks executing within a logical partition are allowed to utilize only the resources assigned to that partition. In the last sentence of paragraph 0023, Armstrong makes it clear that each task is assigned to one of the logical partitions, and hence, can use only the system resources assigned to that partition. Also, in the first sentence of paragraph 0034, Armstrong makes it clear that it is the hypervisor base 202 that enforces logical partitioning. Thus, from these excerpts, it is clear that it is the hypervisor 202 that ensures that tasks executing within a logical partition are allowed to utilize only the resources assigned to that partition. This is in sharp contrast to claim 1 in which it is the operating system that ensures that processes running within the particular non-global operating system partition are allowed to utilize only the resources in the first resource pool. Using an operating system to ensure that tasks executing within a logical partition are allowed to utilize only the resources assigned to that partition is neither disclosed nor suggested by Armstrong.

Overall, Armstrong differs fundamentally from the method of claim 1 in that Armstrong implements the partitioning function at a lower level than claim 1. As made clear in the above arguments, the hypervisor is the component in Armstrong that establishes and enforces the logical partitions. Since the hypervisor executes beneath the OS kernel level (as shown in Fig. 2 of Armstrong), the system of Armstrong implements partitioning at a level lower than the operating system level. In sharp contrast, the method of claim 1 implements partitioning at the operating system level. As a result, it is the operating system that establishes the partitions, enforces the partition boundaries, and ensures that processes running within a partition are allowed to utilize only the resources associated with that partition. Because the two methodologies are implemented at fundamentally different operating levels of a computer system, it should come as no surprise that Armstrong fails to disclose or suggest several aspects of claim 1. For at least the above reasons, Applicants submit that claim 1 is patentable over Armstrong.

Applicants further submit that claims 2-11, which depend from claim 1, and which recite further advantageous aspects of the invention, are likewise patentable over Armstrong for at least the reasons give above in connection with claim 1.

Claim 12

Claim 12 is a machine-readable storage medium counterpart of method claim 1. Applicants submit that claim 12 is patentable over Armstrong for at least the reasons given above in connection with claim 1.

Applicants further submit that claims 13-22, which depend from claim 12, and which recite further advantageous aspects of the invention, are likewise patentable over Armstrong for at least the reasons give above in connection with claim 12.

Claim 23

Claim 23 is an apparatus counterpart of method claim 1. Applicants submit that claim 23 is patentable over Armstrong for at least the reasons given above in connection with claim 1.

Applicants further submit that claims 24-33, which depend from claim 23, and which recite further advantageous aspects of the invention, are likewise patentable over Armstrong for at least the reasons give above in connection with claim 23.

CONCLUSION

For the foregoing reasons, Applicants submit that all of the pending claims are patentable over the art of record, including any art cited but not applied. Accordingly, allowance of all of the pending claims is hereby respectfully solicited.

The Examiner is invited to telephone the undersigned at (408) 414-1080 to discuss any issues that may advance prosecution.

No fee is believed to be due specifically in connection with this Reply. To the extent necessary, Applicant petitions for an extension of time under 37 C.F.R. § 1.136. The Commissioner is authorized to charge any fee that may be due in connection with this Reply to our Deposit Account No. 50-1302.

Respectfully submitted,

Hickman Palermo Truong & Becker LLP

Dated: January 31, 2008

/BobbyKTruong#37499/
Bobby K. Truong
Reg. No. 37,499

2055 Gateway Place, Suite 550
San Jose, California 95110-1089
Telephone No.: (408) 414-1080 ext. 234
Facsimile No.: (408) 414-1076